Tableau Joins

Joins are very important in data analysis in general.

Let's say you have two tables. One is an order table- which shows the orders a customer has made. The other is a product table- which gives you details of the products. And you want to join these two.

Table A: Order Data

order_id	product_id	customer_name	region
1	P1	J. Willows	East
2	P2	P. Jackson	South
3	P3	M. Patel	West
4	P4	R. Davis	West
5	P5	D. Brent	South

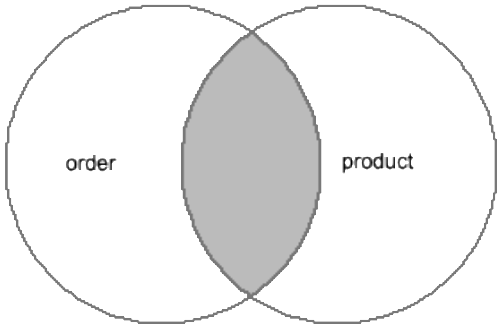
Table B: Product Data

product_id	product_name	category
P1	Pen	Stationery
P2	Phone	Technology
P3	Paper	Stationery
P6	Chair	Furniture

Well tableau gives you four different ways to join this. And we'll go over each one.

Firstly before we start, notice that we can link the two tables by product id as it's present in both.

Inner Join



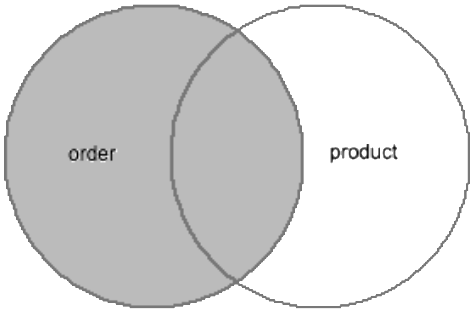
Inner join produces only the set of records that match in both Table A and Table B.

So this is what you'll see on Tableau as your final table

order_id	product_id	customer_name	region	product_id	product_name	category
1	P1	J. Willows	East	P1	Pen	Stationery
2	P2	P. Jackson	South	P2	Phone	Technology
3	P3	M. Patel	West	P3	Paper	Stationery

Because P1, P2, P3 exist in both the tables so that will only pull through.

Left Join



Left join only selects data starting from the left table and matching rows in the right table.

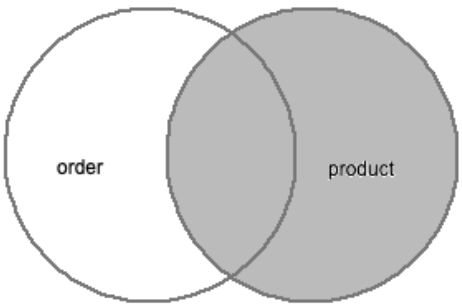
order_id	product_id	customer_name	region	product_id	product_name	category
1	P1	J. Willows	East	P1	Pen	Stationery
2	P2	P. Jackson	South	P2	Phone	Technology
3	P3	M. Patel	West	P3	Paper	Stationery
4	P4	R. Davis	West	NULL	NULL	NULL
5	P5	D. Brent	South	NULL	NULL	NULL

P4 and P5 existed on the orders table but did not exist in the product table. So the final table has NULLs for product data.

P6 existed on your product table but did not exist on the orders table. The left join will bring in all rows from the left table and the matching rows from the right table. So since P6 isn't in the orders table it won't be brought in to your final table.

Right Join

The Right Join is the reverse of left.



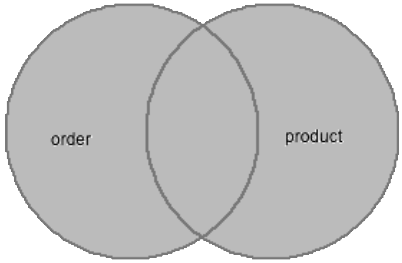
Right join selects data starting from the right table and matching rows in the left table.

order_id	product_id	customer_name	region	product_id	product_name	category
1	P1	J. Willows	East	P1	Pen	Stationery
2	P2	P. Jackson	South	P2	Phone	Technology
3	P3	M. Patel	West	P3	Paper	Stationery
NULL	NULL	NULL	NULL	P6	Chair	Furniture

P6 existed on the product table but did not exist in the orders table. So the final table has NULLs for orders data.

P4 and P5 existed on the order table but did not exist on the product table. The right join will bring in all rows from the right table and the matching rows from the left table. So since P4 and P5 isn't in the product table it won't be brought in to your final table.

Full Outer



A full outer join has a final set that contains all rows from both left and right tables, with the matching rows from both sides where available. In case there is no match, the missing side will have NULL values.

order_id	product_id	customer_name	region	product_id	product_name	category
1	P1	J. Willows	East	P1	Pen	Stationery
2	P2	P. Jackson	South	P2	Phone	Technology
3	P3	M. Patel	West	P3	Paper	Stationery
4	P4	R. Davis	West	NULL	NULL	NULL
5	P5	D. Brent	South	NULL	NULL	NULL
NULL	NULL	NULL	NULL	P6	Chair	Furniture

Now full outers aren't always available. Generally if you're connecting to databases of different types, its greyed out. For mysql databases, it's also greyed out.